

TITLE OF THE INVENTION

ADDRESS SEARCH APPARATUS AND METHOD IN ETHERNET SWITCH

CLAIM OF PRIORITY

This application makes reference to, incorporates the same herein, and claims all benefits accruing under 35 U.S.C. §119 from my application *ADDRESS SEARCH APPARATUS AND METHOD IN ETHERNET SWITCH* filed with the Korean Industrial Property Office on 22 November 1999 and there duly assigned Serial No. 51825/1999.

BACKGROUND OF THE INVENTION

Field of the Invention

The invention relates to a data network, and more particularly an apparatus and a method for performing an efficient address search in a switch network.

Description of the Related Art

Ethernet is a most typical protocol of LAN(Local Area Network) which occupies a very important portion in data networks. Ethernet is widely accepted in the industry already, and the application thereof reaches all kinds of data networks. At the present, since users are rapidly increasing and more various services are getting provided, bandwidth of a network is required to be wider than before. Providing wide bandwidth of a network enables converting from a repeater so called hub into a switch in a remarkably rapid speed. As compared to a repeater which shares a

1 transmission medium, a switch efficiently uses only required sources from transmission means to
2 receiver means thereby having a very rapid transmission speed.

3 Herein, "address search" means that an ethernet switch establishes required information to
4 determine an output terminal, and collects and utilizes information for address. The address search
5 is particularly important in an application in which management for address is highly required.
6 Examples of the application include ARP(Address Resolution Protocol) management,
7 NMS(Network Management System), RMON(Remote Monitoring) and etc.

8 In an example of an address searching method in an ethernet switch as shown in FIG. 1, a
9 search memory interface(SMI) 6 receives a header and a port number of an input packet from a
10 PCU(Port Control Unit) in each of port devices 2-0, 2-1, ..., 2-7 to establish data about source
11 address on a search memory, and determines an output port of each packet on the basis of the
12 establish data.

13 However, in Fig. 1, since the 8 port devices are required to determine output ports by using
14 the same search unit, the input packet of one port (device) only can be searched while the remaining
15 ports (devices) are left to wait. Also, there are some structural problems in realizing a search logic
16 required for a management switch, i.e., a switch which has a control unit for controlling an ethernet
17 switch when the switch structure is a non-management switch, i.e. a switch which does not have a
18 control unit for controlling an ethernet switch. An interface for sharing address data with other units
19 was not defined also. In other words, in the non-management switch structure, the address structure
20 is not suitably realized for a management switch.

SUMMARY OF THE INVENTION

Therefore, it is an object of the present invention to provide an improved ethernet switch and switching process.

It is also an object of the present invention to provide an apparatus and a method for performing an efficient address search operation in an ethernet switch network.

It is another object of the present invention to provide an apparatus and a method in which conversion from a non-management switch structure into a management structure or a port expansion is easy.

According to the foregoing objects of the present invention, it is provided with a processing structure in which a number of ports use one search unit, said structure having a local search block corresponding to each of the ports to perform a search operation as a distributed processing structure.

It is also provided with a structure in which an information exchange between the local search blocks and a main search block for port and a search operation on individual addresses can be proceeded without an error.

Furthermore, it is provided with an address search apparatus in an ethernet switch having a plurality of ports, having a local search unit corresponding to each of the plurality of ports; wherein the local search unit includes: a plurality of local search blocks for analyzing a destination address of an input packet received in the port thereof to search for a transmission port of the ethernet switch and provide a source address thereof; a main search unit for analyzing the source address to establish an address data structure of the plurality of local search blocks, and for responding to a destination address request from at least one of the plurality of local search blocks to provide the requested

1 destination address to a corresponding local search block by using the address data structure; and a
2 scheduler for controlling the local search blocks and the main search unit to enable an interface
3 therebetween.

4 BRIEF DESCRIPTION OF THE DRAWINGS

5 A more complete appreciation of the invention, and many of the attendant advantages thereof,
6 will be readily apparent as the same becomes better understood by reference to the following detailed
7 description when considered in conjunction with the accompanying drawings in which like reference
8 symbols indicate the same or similar components, wherein:

9 FIG. 1 is a view for showing an address searching method in an ethernet switch;

10 FIG. 2 is a view for showing an address searching method in an ethernet switch according
11 to an embodiment of the present invention;

12 FIG. 3 is a detailed block diagram for showing one of local search blocks shown in FIG. 2;

13 FIG. 4 is a detailed block diagram of a main search block shown in FIG. 2;

14 FIG. 5 is a message format diagram for an interface between search blocks according to an
15 embodiment of the present invention; and

16 FIG. 6 is a flow chart illustrating how an incoming packet of information is sent to a
17 destination port according to an embodiment of the present invention.

18 DETAILED DESCRIPTION OF THE INVENTION

19 Referring to FIG. 1 in more detail, each of the port devices 2-0, 2-1, ..., 2-7, which

corresponds to each of ports 0, 1, ..., 7 for example, includes a physical hierarchy unit for functioning as a physical hierarchy, a MAC(Medium Access Control) unit for functioning as a medium access control, and a PCU(Port Control Unit). When a packet is received by at least one port at random, some of the port devices 2-0, 2-1, ..., 2-7 of the ports corresponding to the received packet, output control signal command availables(cmd_avl) to a scheduler 4, respectively. When receiving control signal cmd_avl, the scheduler 4 which is periodically operated controls one port to be selected via competition and enables an address search about the selected one port (device). In other words, when receiving the control cmd_avl, the scheduler 4 schedules a priority set on the basis of a predetermined control or a port selection order on the basis of a round robin, and provides a resultant port selection signal to both of a multiplexer(MUX) 10 and a demultiplexer(DEMUX) 12. The predetermined control can be set by users for example. The multiplexer 10 selects one from 8 port devices 2-0, 2-1, ..., 2-7) in response to the port selection signal provided by the scheduler 4, and receives a header and a port number from the selected port device to output to search memory interface 6. The header can include destination address, source address, and etc. Then, the search memory interface 6 establishes data about the source address on the search memory 8, and determines an output port corresponding to the selected input packets on the basis of the established data. After determining the output port, the search memory interface 6 outputs the determined information to the corresponding one of the 8 port devices 2-0, 2-1, ..., 2-7 via the demultiplexer 12. After that, the search memory interface 6 outputs a control signal cmd_cpl which notifies completion of output port determination to the scheduler 4.

FIG. 2 is a view for showing an address searching method in an ethernet switch according

to an embodiment of the present invention. Referring to FIG. 2, an address searching apparatus according to an embodiment of the present invention includes 8 local search blocks 20-0, 20-1, ..., 20-7, for example, and one main search block 26, in which an interface protocol is defined for information exchange between the local search blocks 20-0, 20-1, ..., 20-7 and the main search block 26. Each of the 8 local search blocks 20-0, 20-1, ..., 20-7 is arranged to correspond to each of 8 port devices 22-0, 22-1, ..., 22-7 arranged within the ethernet switch, and the main search block 26 is arranged between the scheduler 24 and a search memory (not shown in FIG. 2). Each of the 8 local search blocks 20-0, 20-1, ..., 20-7 analyzes a destination address of an input packet received in a port thereof to search for a transmission port of the ethernet switch, and the main search block 26 analyzes the destination address to establish an address data structure of the local search blocks. When transmitting and receiving data, the scheduler 24 which is periodically operated permits a port to be selected from the plurality of ports and interfaces control signals between the local search blocks 20-0, 20-1, ..., 20-7 and the main search block 26.

FIG. 3 is a detailed block diagram for showing one of local search blocks 20-0, 20-1, ..., 20-7 shown in FIG. 2. Referring to FIG. 3, each of the local search blocks 20-0, 20-1, ..., 20-7 has 2 address tables, i.e., a destination address table 30 and a source address table 32. In the destination address table 30, destination address and destination information (such as transmission port numbers) are matched. In the source address table 32, source addresses and source information (such as input port numbers) are matched. Also, address sorting logics 34 and 36 corresponding to the destination address table 30 and the source address table 32 respectively are arranged to sort an ethernet address into groups as many as necessary. Each of the local search blocks 20-0, 20-1, ..., 20-

7 has a control logic 38 for controlling the corresponding search block and a register unit 40 for temporal storage of data.

FIG. 4 is a detailed block diagram of the main search block 26 shown in FIG. 2. The main search block 26 includes an address table 50, a table access logic 48, an address sorting logic 46, a control logic 42, and a register unit 44. The address table 50 is a memory for storing addresses known to a system. In using the memory, a hash function ("h") is used to satisfy characteristics of the ethernet address. The hash function can be subdivided according to methods for obtaining hash keys, in which a method using XOR gate(exclusive OR gate) is adapted in an embodiment of the present invention. The table access logic 48 is a kind of access logic for accessing the address table 50 structured as above. The table access logic 48 performs the foregoing hash function. Therefore, when the individual addresses are searched, desired information stored in the address table 50 can be obtained via the table access logic 48. The address sorting logic 46 is a logical function block for classifying addresses having same characteristics in order to store data known to the system into the address table 50. The control logic 42 and the register unit 44 have blocks, i.e., an external processor and an interface for controlling each of these blocks according to functions and for determining conditions thereof. Also, the control logic 42 and the register unit 44 exchange information via an interface between the local search blocks 20-0, 20-1, ..., 20-7 corresponding to each of the port device 22-0, 22-1, ..., 22-7, and construct each address table.

FIG. 5 is a message format diagram for an interface between the search blocks according to the embodiment of the present invention. Referring to FIG. 5, messages include a command field CMD, an ID field(identifier field), and a specification information field according to the CMD

information. In the command field CMD, commands(CMD) such as search, add, delete and update can be recorded, and the ID field includes a transmitter ID for transmitting the messages of FIG. 5 and a receiver ID for receiving the messages. For example, when transmitting messages from the local search block 20-0 to the main search block 26, the transmitter ID will be the ID of the local search block 20-0 and the receiver ID will be the ID of the main search block 26. The specification information field according to the CMD information includes specification information according to the CMD information. The CMD information may include search, add, delete, update and etc. as described above, and the resultant specification information may include MAC address, header information of a packet, port ID, device number, input port number and etc. The header information of a packet includes source and destination addresses.

Hereinafter, operations for address search according to the embodiment of the present invention will be described in more detail in reference to FIG. 2 to FIG. 5 in conjunction with FIG. 6. When a packet is received (Step 100) in a certain port device 22-N (herein, N is one of 0 to 7) of the 8 port devices 22-0, 22-1, ..., 22-7, the local search block 20-N corresponding to the port device 22-N reads a destination address from a header information (Step 105) of the packet to compare with the destination addresses in the destination address table 30 (Step 110). The foregoing comparing is performed by the address sorting logic 34. The address sorting logic 34 performs address sorting, i.e., an operation for classifying an ethernet address into groups as many as necessary, thereby more easily accessing the internal destination table 30. An example of algorithms for classifying into groups may include a hash algorithm. When the destination address of the input packet exists in the address table 30, the control logic 38 of the local search block 20-N obtains a required information

1 (Step 115), i.e., a destination information (transmission port). After that, the local search block 20-N
2 rejects the input packet via filtering when the destination information (transmission port) is proved
3 to be its own port (Step 125). Describing the above in more detail, when obtaining the destination
4 information (transmission port), the address sorting logic 36 compares the source addresses in the
5 source address table 32 with the destination address of the input packet (Step 120). When the
6 destination address of the input packet exists in the source address table 32, the input packet is
7 rejected by filtering (Step 125). In other words, the packet is proved that transmission to other port
8 is not required, and then filtered. In turn, when the destination information (transmission port)
9 obtained from the destination address table 30 is not identical with the destination address of the
10 input packet in the source address table 32, or when the destination information (transmission port)
11 obtained from the destination address table 30 is a different port, the address sorting logic 36
12 provides the corresponding information to the corresponding port device 22-N.

13 In turn, when the destination address of the input packet does not exist in the address table
14 30 in the local search block 20-N (Step 110), the address sorting logic 36 receives a corresponding
15 destination address from the main search block 26 to update the destination address table 30 in the
16 local search block 20-N. Describing the above in more detail, the local search block 20-N sends a
17 message loaded with a search command in the command field CMD, a source ID and a destination
18 ID in the identifier field ID, and header information of the packet in the specification information
19 field according to the CMD information to the main search block 26 (Step 130), according to the
20 protocol defined between the local search blocks 20-0, ..., 20-7 and the main search block 26. Then,
21 the main search block 26 performs an internal search (Step 135) about the destination address in the

1 received header information, and when the address search is successfully completed, transmits an
 2 information about the searched destination address to the corresponding local search block 20-N.
 3 Then, the corresponding local search block 20-N transmits the destination information (transmission
 4 port) received from the main search block 26 to the corresponding port device 22-N thereby
 5 permitting the port device 22-N to transmit the input packet (Step 160), and updates the table
 6 information of the destination address table 30 (Step 155) in the local search block 20-N at the same
 7 time. If an information of the corresponding destination address (hereinafter referred to "port
 8 information") does not exist in the main search block 26 in addition to being found in the local
 9 search block, the main search block 26 sends a no-port information of the destination to the
 10 corresponding local search block 20-N (Step 145) since the port information about the address does
 11 not exist in the present system. When the corresponding local search block 20-N receives the no-port
 12 information, the corresponding port device 22-N performs a forwarding operation. The forwarding
 13 operation means broadcasting (Step 150). In other words, since it is not determined to which port
 14 the input packet is required to be transmitted, the packet is transmitted to all available ports (except
 15 input ports) in the ethernet switch. In turn, the local search blocks 20-0, 20-1, ..., 20-7 corresponding
 16 to the ports respectively and the main search block 26 periodically perform an update operation of
 17 the source and destination addresses.

18 Hereinafter, the operation will be described in detail. The local search blocks 20-0, 20-1, ...,
 19 20-7 corresponding to the ports respectively periodically transmit header information (such as source
 20 addresses, destination addresses and etc.) to the main search block 26 so that the main search block
 21 26 can perform an address update (Steps 143 & 157). The main search block 26 establishes an

1 address data structure via address learning via input of the header information (such as source
2 addresses, destination addresses and etc.) transmitted from the local search blocks 20-0, 20-1, ..., 20-
3 7 which correspond to the ports respectively. In other words, the information used for establishing
4 the address data are port information of the sub hosts of the system and addresses allocated thereto.
5 This information can be obtained since each of the hosts causes its address information to be
6 included in the header of the packet while transmitting and receiving the packets. The information
7 corresponding to the source address, as though it may be varied according to objects to which the
8 system is aimed, can be supposed simply as a port information in the system (i.e., as to which port
9 the source address corresponds).

10 Describing the address learning process in more detail, the main search block 26 searches
11 whether a corresponding information (such as address information, port information and etc.) exists
12 in the address table 50 of the main search block 26 about the destination and source addresses. When
13 a corresponding information exists, the main search block 26 performs an information update
14 operation desired by the system including an address aging. Herein, the address aging means clearing
15 old address data, i.e. address data which are not used for a predetermined period, from the addresses
16 stored in the address table 50 of the main search block 26 (Step 157). Performing the information
17 operation including the address aging like this can establish new information as address data. When
18 a corresponding information does not exist, the address learning operation is performed. The address
19 learning is generally performed by an external processor, and data insertion is carried out into the
20 link structure by using hash within the main search block 26 (Step 143). In other words, the search
21 information (such as MAC address, port number and add command) connected in respect to the

1 corresponding source address is provided by the external processor, but actual operation to connect
2 data to the link according to the hash key is carried out in the main search block 26.

3 In turn, the local search block 20-N requests port information of the destination and source
4 addresses to the main search block 26 when the destination and source addresses of the input packets
5 do not exist in the destination table 30 and the source address table 32 of the local search block 20-N.
6 Then, the main search block 26 provides port information of the requested destination address and
7 source addresses to the local search block 20-N so that the local search block 20-N may update the
8 destination and source addresses and corresponding port information in the destination address table
9 30 and the source address table 32.

10 As described above, the invention can reduce required time in a large amount by managing
11 a small table for each port than search operation which are carried out via competition while sharing
12 single search logic. Therefore, the main search block can carry out more tasks. Also, exchange of
13 address information with other system can be carried out by conversion into a management switch
14 structure or a port expansion, and the overall search efficiency can be enhanced by management of
15 suitable table by each port.

16 While the invention has been described in detail by specific reference to preferred
17 embodiments thereof such as an ethernet switch, it is understood that variations and modifications
18 thereof may be made without departing from the true spirit and scope of the invention. In other
19 words, the variations and modifications can be made to a switch network which include at least one
20 of packet switch, asynchronous transfer mode (ATM) network and etc. Therefore, it should be
21 understood that the true spirit and the scope of the invention are not limited by the above

embodiment, but defined by the appended claims and the equivalents thereof.

[illegible]